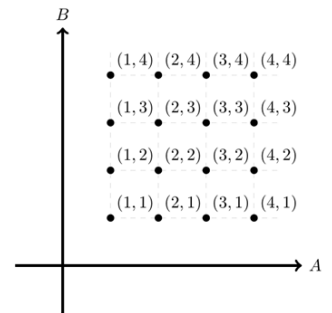
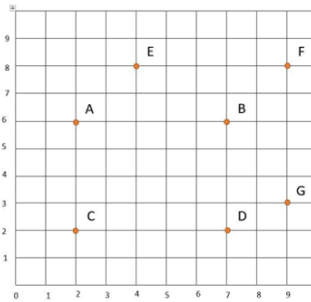
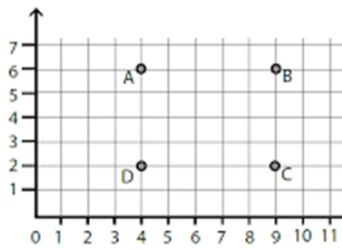


### Actividad 6 (Slam de LiDAR)

#### Instrucciones:

1. Crear un nuevo repositorio nuevo con el nombre: Actividad 6 (SLAM de LiDAR)
2. Implementar el código requerido para generar el seguimiento de los siguientes waypoints de forma aleatoria, ajustando los parámetros: `sampleTime`, `tVec`, `initPose`, `lidar.scanAngles`, `lidar.maxRange`, `waypoints`, `controller.LookaheadDistance`, `controller.DesiredLinearVelocity` y `controller.MaxAngularVelocity`. Evadiendo los obstáculos del mapa de navegación “exampleMap”



- 3.
4. Implementar el código requerido para generar el seguimiento de los siguientes waypoints de forma secuencial: (1, 2), (2, 10), (11, 8), (8, 2), (8, 8) y (1, 2) ajustando los parámetros: `sampleTime`, `tVec`, `initPose`, `scanAngles`, `lidar.maxRange`, `waypoints`, `controller.LookaheadDistance`, `controller.DesiredLinearVelocity` y `controller.MaxAngularVelocity`. Evadiendo los obstáculos del mapa de navegación “exampleMap”

#### Introducción:

La tecnología LiDAR (Light Detection and Ranging) se ha convertido en una herramienta fundamental para la percepción y mapeo en entornos dinámicos y complejos. Su uso en aplicaciones como la robótica, la conducción autónoma y la navegación ha permitido el desarrollo de soluciones cada vez más precisas y eficientes.

Una de las aplicaciones más importantes de LiDAR es en la construcción de mapas y localización simultánea (SLAM, por sus siglas en inglés), que permite a un robot o vehículo autónomo moverse en un entorno desconocido y crear un mapa detallado del mismo. El objetivo de SLAM es determinar la posición y orientación del vehículo mientras se crea el mapa.

En esta tarea, se explorará cómo se lleva a cabo SLAM utilizando LiDAR y cómo se pueden resolver los desafíos asociados con esta técnica. También se analizarán las diferentes técnicas y algoritmos utilizados en el proceso de SLAM y se discutirán las limitaciones y áreas de mejora en esta área de investigación.

En resumen, esta tarea tiene como objetivo proporcionar una visión general de SLAM con LiDAR y explorar sus aplicaciones y desafíos actuales en el campo de la robótica y la conducción autónoma.

### **Desarrollo:**

En este código se muestra cómo un robot diferencial puede seguir una trayectoria definida por una serie de waypoints y esquivar los obstáculos que se encuentre en su camino. Para ello, se emplean los algoritmos de Pure Pursuit y Vector Field Histogram (VFH), que controlan el movimiento del robot a partir de un modelo matemático. El código simula el comportamiento del robot en un mapa que representa el entorno, donde se fija la posición inicial del robot y los puntos de referencia que debe alcanzar. En cada iteración del bucle de simulación, se actualiza la trayectoria del robot y se usa un lidar para medir las distancias y evitar colisiones. Así, el código simula la navegación autónoma de un robot diferencial que combina dos algoritmos para seguir waypoints y evitar obstáculos.

```
%% EXAMPLE: Differential Drive Path Following
% In this example, a differential drive robot navigates a set of waypoints
% using the Pure Pursuit algorithm while avoiding obstacles using the
% Vector Field Histogram (VFH) algorithm.
%
% Copyright 2019 The MathWorks, Inc.
%% Simulation setup
% Define Vehicle
R = 0.1; % Wheel radius [m]
L = 0.5; % Wheelbase [m]
dd = DifferentialDrive(R,L);
% Sample time and time array
sampleTime = 0.1; % Sample time [s]
tVec = 0:sampleTime:120; % Time array
% Initial conditions
initPose = [1;1;-90]; % Initial pose (x y theta)
pose = zeros(3,numel(tVec)); % Pose matrix
pose(:,1) = initPose;
% Load map
close all
load exampleMap
% Create lidar sensor
lidar = LidarSensor;
lidar.sensorOffset = [0,0];
lidar.scanAngles = linspace(-pi/2,pi/2,51);
lidar.maxRange = 2;
% Create visualizer
viz = Visualizer2D;
```

```

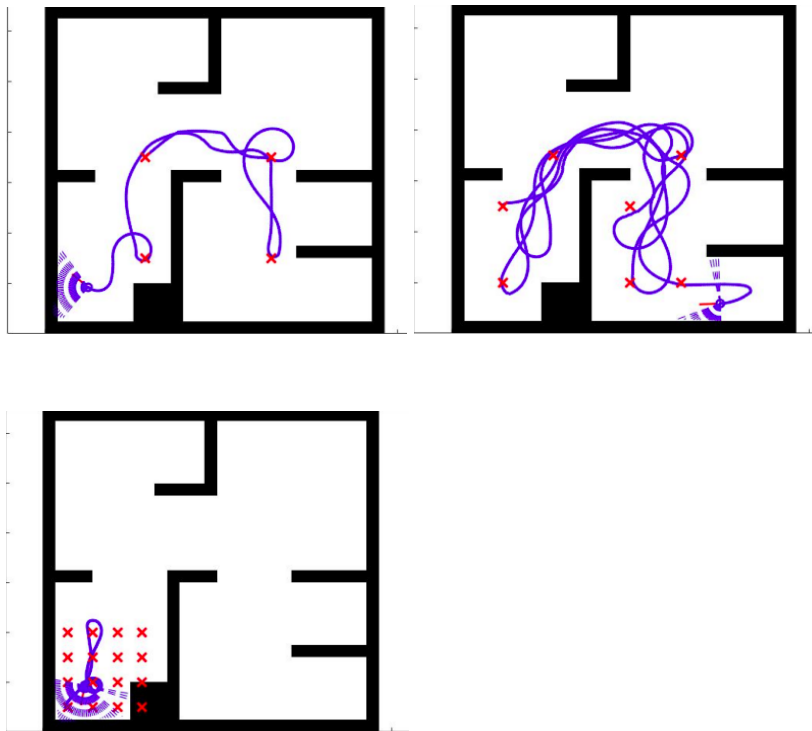
viz.hasWaypoints = true;
viz.mapName = 'map';
attachLidarSensor(viz,lidar);
%% Path planning and following
% Create waypoints
waypoints = [1,1; 1,2; 1,3; 1,4; 2,1; 2,2; 2,3; 2,4; 3,1; 3,2; 3,3; 3,4; 4,1;
4,2; 4,3; 4,4];
% Pure Pursuit Controller
controller = controllerPurePursuit;
controller.Waypoints = waypoints;
controller.LookaheadDistance = 0.2;
controller.DesiredLinearVelocity = 0.3;
controller.MaxAngularVelocity = 30;
% Vector Field Histogram (VFH) for obstacle avoidance
vfh = controllerVFH;
vfh.DistanceLimits = [0.05 3];
vfh.NumAngularSectors = 36;
vfh.HistogramThresholds = [5 10];
vfh.RobotRadius = L;
vfh.SafetyDistance = L;
vfh.MinTurningRadius = 0.25;
%% Simulation loop
r = rateControl(1/sampleTime);
for idx = 2:numel(tVec)
% Get the sensor readings
curPose = pose(:,idx-1);
ranges = lidar(curPose);
% Run the path following and obstacle avoidance algorithms
[vRef,wRef,lookAheadPt] = controller(curPose);
targetDir = atan2(lookAheadPt(2)-curPose(2),lookAheadPt(1)-curPose(1)) -
curPose(3);
steerDir = vfh(ranges,lidar.scanAngles,targetDir);
if ~isnan(steerDir) && abs(steerDir-targetDir) > 0.1
wRef = 0.5*steerDir;
end
% Control the robot
velB = [vRef;0;wRef]; % Body velocities [vx;vy;w]
vel = bodyToWorld(velB,curPose); % Convert from body to world
% Perform forward discrete integration step
pose(:,idx) = curPose + vel*sampleTime;
% Update visualization

```

```
viz(pose(:,idx), waypoints, ranges)
waitfor(r);
end
```

### **Resultados:**

Los resultados obtenidos son producto de las diferentes pruebas que se hicieron con diferentes valores, como la distancia dentro de un muro, la velocidad angular, y el tiempo de muestreo que permiten mejorar la ruta que sigue nuestro dispositivo



### **Conclusión:**

La tecnología LiDAR se ha convertido en una herramienta fundamental para la percepción y mapeo en entornos dinámicos y complejos. La técnica de localización simultánea y mapeo (SLAM) con LiDAR permite a los robots y vehículos autónomos moverse en un entorno desconocido y crear un mapa detallado del mismo mientras se determina la posición y orientación del vehículo. Aunque existen desafíos y limitaciones, se espera que la tecnología y los enfoques continúen mejorando para permitir la creación de soluciones más precisas y eficientes en el futuro.

### **Bibliografía:**

Ignacio. (2022). LiDAR: principio de funcionamiento y principales aplicaciones. Pyroistech. <https://www.pyroistech.com/es/lidar-es/>